Automatic Content Extraction of Filled Form Images Based on Clustering Component Block Projection Vectors

Hanchuan Peng [†‡], *Xiaofeng He* [‡], *and Fuhui Long* [*]

[†] *Computer Science and Mathematics Division, Oak Ridge National Lab.*
Office: A110 Life Science Building, 120 Green St., Athens, GA, 30605.
Email: penghanchuan@yahoo.com. http://www.hpeng.net

[‡] *Computational Research Division, Lawrence Berkeley National Laboratory,*
University of California, Berkeley, CA, 94720.
Email: xhe@lbl.gov.

[*] *Center for Cognitive Neuroscience, Duke University, Durham, NC, 27710.*
Email: long@neuro.duke.edu.

ABSTRACT

Automatic understanding of document images is a hard problem. Here we consider a sub-problem, automatically extracting content from filled form images. Without pre-selected templates or sophisticated structural/semantic analysis, we propose a novel approach based on clustering the component-block-projection-vectors. By combining spectral clustering and minimal spanning tree clustering, we generate highly accurate clusters, from which the adaptive templates are constructed to extract the filled-in content. Our experiments show this approach is effective for a set of 1040 US IRS tax form images belonging to 208 types.

Keywords: Document analysis, Image classification, Form processing, Image understanding, clustering.

1. INTRODUCTION

Document image processing and analysis are important techniques in office automation and digital library applications. One hard problem is how to automatically extract and annotate filled-in content of form images [1]. An intuitive approach is to analyze the structure of an input form image [16] using only local information such as lines [15], intersections [4], cells [14], etc. The difficulty of this approach is that the structural information could be inaccurate, due to the poor quality of form images (e.g. touching of filled-in content and original form-texts, deformation or information loss in image acquisition, etc.). In addition, the form structure could be very complicated, making it hard to "understand" the meaning of individual regions. Another semantic analysis approach is to extract and discriminate information of different types [7], based on techniques of image segmentation, graphics/text separation, language separation, text recognition, etc. This paradigm, however, often has

Document Recognition and Retrieval XI, edited by Elisa H. Barney Smith, Jianying Hu, James Allan,
Proc. of SPIE-IS&T Electronic Imaging, SPIE Vol. 5296 © 2004 SPIE and IS&T · 0277-786X/04/$15

difficulty in distinguishing the filled-in and the original content of the form image, especially when they have touching parts.

An alternative way to attack this problem is to use the Document-Image-Recognition (DIR) methods. The goal of DIR is to automatically determine the types of input document images. Template matching methods such as interval-code-matching [5], CCITT G4 pass-code matching [6], component-block-list matching [9], component-block-projection matching [8], etc, have been proposed. If the type of an input form image can be classified accurately, the template-image (or pre-stored content description) for this form can be used to differentiate the filled-in content from the original form text/graphics. In this case, the quality of pre-selected templates is a critical factor.

Unfortunately, it is often difficult to select single good template for an archive of filled form images, due to the great variations of filled-in content. In some other situations, form-templates might not have been pre-selected, or it is not applicable to select them manually (e.g. due to the expensive manual classification of hundreds of form types).

In this paper, we propose a new approach for automatic content extraction of filled form images of unknown types, without pre-selected templates. The approach is based on form image clustering instead of complicated structure or semantic analysis. The similarity between images is calculated based on their component-block-projection-vectors. We develop a high-performance spectral clustering scheme to split the form image samples into many coherent clusters. Then, an adaptive template for each cluster is generated using the corresponding filled form images in this cluster. We show the adaptive templates thus found well approximate the original non-filled (blank) form images. Therefore, by subtracting the constructed template from an input form image, we are able to extract the filled-in content of the form. This also facilitates further automatic annotation of the filled-in content via semantic analysis of template content.

2. CBP REPRESENTATION OF FORM IMAGES

It is known that the error is large when form images are directly matched, because the filled-in information has large impact on the similarity of two images. To overcome this problem, we have previously suggested using the component blocks [9][8], i.e., the boundaries of the content regions (texts or graphics) of a form image. Because two forms of the same type have similar input fields, they should look alike if the filled-in content is removed. In the extreme case, all (i.e. both the filled-in and the original) content in a form image can be excluded, leaving only the bounding boxes (i.e. component blocks) of the isolated content regions. This component-block-image is a binary image made up of rectangular boxes, where foreground pixels (i.e., box edges) take value 1 and background pixels take value 0. It can be written as an array

$$\begin{bmatrix} b_{11} & \cdots & b_{1w} \\ \vdots & b_{mn} & \vdots \\ b_{h1} & \cdots & b_{hw} \end{bmatrix} \quad (1 \le m \le h;\ 1 \le n \le w), \tag{1}$$

where w and h are the image width and height, respectively; $b_{mn}=1$ if the pixel $\{m,n\}$ is on the edge of a component block, otherwise $b_{mn}=0$.

Usually, the component-block-images of the same type should have larger overlap than the component-block-images of different types. To match component-block images, a simple and robust scheme, i.e., the Component-Block-Projection (CBP) vectors, was suggested in [8]. A CBP vector is the concatenated directional (horizontal and vertical) projection vector of all component blocks in a

form image, as shown in Eq. (2). Hence, each form image is represented as a point in the space of CBP vectors. This feature provides means to canonical and efficient matching of form images.

$$[\sum_{n=1}^{w} b_{1n}, \cdots, \sum_{n=1}^{w} b_{hn}, \sum_{m=1}^{h} b_{m1}, \cdots, \sum_{m=1}^{h} b_{mw}]' \quad (2)$$

3. CLUSTERING FORM IMAGES

We compute the similarity scores of pairwise CBP vectors. Suppose there are N document-image samples, we obtain an $N \times N$ similarity matrix, based on which we cluster the form images of unknown types.

Three simple similarity scores, i.e. correlation coefficient, L_1 similarity, and L_2 similarity, are considered. The correlation coefficients are normalized to the range [0, 1]. The L_1 similarity is defined as the exponent of the negative L_1 distance, thus having range [0, 1]. The L_2 similarity is defined in the same way.

A simple clustering method using similarity matrix is based on Minimal Spanning Tree (MST). The idea is to construct a max-weight MST from the similarity matrix, then iteratively remove the most dissimilar tree-edges (edges with the smallest weight/similarity) until a preset number of clusters are generated (or some constraints on the found clusters are satisfied). One problem of MST-clustering, however, is that the global information of the similarity matrix has not been well considered; consequently, usually many leafs are cut, resulting in bad clusters.

The data (similarity matrix) can also be clustered using spectral clustering methods [10][13][3], which use the top eigenvectors of the similarity matrix as the indicators of different clusters. This approach is able to generate globally more coherent clusters, so that the data points in the same cluster are more similar than those belonging to different cluster. The spectral clustering method can be iteratively applied to generate hierarchical clusters. One potential problem of spectral clustering is that low-value similarities might be noisy. Hence, a threshold is usually needed to remove low-value entries and improve the clustering.

We try to make use of the advantages of both MST and spectral clustering, and avoid their problems. We propose a simple method to apply spectral clustering to the MST induced from the similarity matrix, to generate globally meaningful clusters. In this way, the similarity matrix becomes extremely sparse, which enables the spectral clustering to handle very large number of samples (form images), in addition to the benefit of noise-reduction. Because the expected number of clusters (types of form images) is usually large (tens to hundreds), another concern is that using the very sparse similarity matrix can accelerate the clustering procedure.

In our implementation, we treat each document-image sample as a graph node. The similarity of each pair of nodes is set to be the weight of the respective edge between these nodes. We use the Prim's algorithm [2] to construct MST from the fully connected graph. Since the number of undirected edges is $|E| = (N^2 - N)/2$, the MST construction has the complexity $O(|E| \ln N)$ when we use the ordinary binary heap and can be reduced to $O(|E| + N \ln N)$ when we use Fibonacci heap [2]. The found MST is an undirected graph; the respective adjacency matrix, denoted as T, is symmetric.

In our spectral clustering method, we add an $N \times N$ identity matrix I_N onto T and produce the following symmetric matrix M, indicating the self-similarity of every document-image is the strongest:

$$M = T + I_N \quad (3)$$

We assume the clustering results are faithful, i.e. the form images clustered together have the same true type. This assumption puts a great requirement on the quality of the clusters induced.

However, as we will see from the experimental results, the above MST based spectral clustering can generate satisfactory results.

4. TEMPLATE-FORM CONSTRUCTION AND CONTENT EXTRACTION

We construct a template image for each cluster of form images. Clearly, in a generative model, we can assume all form images of the same type have an *unknown* independent identical distribution (i.i.d.). Based on the weak law of large numbers, the sample mean value will converge to the true distributional expectation, i.e. the centroid of the unknown distribution, given a sufficient number of form images.

We note that the centroid of the filled form images is not necessarily a good template for content extraction, because the filled-in information will also appear on this Centroid Form Image (CFI). To solve this problem, we produce the pairwise difference images of samples in a cluster. Because of the assumption of highly accurate clusters, the original (non-filled-in) information on the form images would be removed in the difference image (we use the absolute value of the difference). Therefore, by further assuming the difference images are i.i.d., we can use the mean difference images to approximate the Centroid Difference Image (CDI). The foreground regions of CDI correspond to the most probable regions of filled-in content.

We use the detected non-filled form images as templates for the respective image clusters. The template image can be obtained by subtracting the CDI from CFI. It is possible that its quality is not very satisfactory (e.g. fuzzy edges, broken lines, etc). By considering some image aligning schemes, the quality of the template image can be improved.

The filled-in content can be extracted by subtracting the template image (non-filled form) from each filled form image. However, any overlapping pixels of the two images will be removed. Thus, we use a simple pixel repair process based on the neighborhood information of every subtracted pixel: a pixel will not be subtracted from the form image if the majority of its $m \times n$ neighboring pixels is not removed. In this stage, the image aligning is also employed to avoid fuzzy content boundary of the non-filled form images.

For a complete form-image content-understanding system, it also needs to consider subsequent problems such as automatic annotation of filled-in content. Given the accurate template images constructed, this can be accomplished by using character-recognition engines and word-sequence-aligning/matching methods.

5. EXPERIMENTS

We used a dataset from [8], USTAX208, contains 1040 filled tax form images of the USA Internal Revenue Service (IRS) (http://www.irs.gov/formspubs/index.html). These forms belong to 208 different types (5 forms per type). Due to the great content variations on these forms, it is difficult to directly use a structural/semantic analysis to extract the filled-in content. Along with the CBP representation approach, each form image was normalized to 800×600 pixels, based on which the component blocks were extracted using the PageX package [7]. The area of each block was used to indicate the importance of the block. For each image, we only used the K largest blocks to generate the CBP vectors.

5.1. Clustering Results

Knowing that there are 208 ground truth clusters of the 1040 form images, we used the MinMaxCut scheme [3] of spectral clustering to generate 208 clusters for comparison. The *F*-measure [11] between the clustering results and the ground truth clusters was used to evaluate the quality of results. As shown in Eq.(4), *F*-measure can be written in the form of precision *P* (Appendix Eq.(A1)) and recall *R* (Appendix Eq.(A2)), and has the range [0, 1]. The larger the *F*-measure, the better the clustering results. Usually, *F*-measure has a value close to the accuracy.

$$F = 2 \cdot \frac{P \cdot R}{P + R} \tag{4}$$

Table 1. The F-measure between clustering results and the ground truth. K is the number of component blocks used in generating the CBP vectors. For the spectral clustering columns, F(α) is the F-measure value obtained by thresholding at α, which takes value 0 (i.e. *no thresholding*), μ (*mean*), μ+σ (*std*), *and* μ+*2*σ. *For the MST clustering columns, F*(β) *is the F-measure of results obtained through the common MST clustering (termination criterion to split a cluster: maximal-edge-weight ≤ β × minimal-edge-weight). F(MST-Spectral) is the F-measure of results obtained via combining spectral clustering and MST clustering.*

Similarity	*K*	*Spectral Clustering*				*MST Clustering*			*F* (MST-Spectral)
		F (0)	*F* (μ)	*F*(μ+σ)	*F*(μ+2σ)	*F*(1.5)	*F*(1.2)	*F*(1.05)	
Correlation coefficient	10	0.9074	0.9322	0.9505	0.9790	0.9829	0.9330	0.7947	1.0000
	20	0.9198	0.9375	0.9532	0.9951	0.9923	0.9642	0.8340	1.0000
	30	0.9068	0.9128	0.9514	0.9981	0.9956	0.9686	0.8392	1.0000
	40	0.9290	0.9180	0.9693	0.9990	0.9951	0.9717	0.8440	1.0000
	50	0.9313	0.9038	0.9664	0.9937	0.9956	0.9734	0.8446	1.0000
	60	0.9313	0.8987	0.9660	0.9966	0.9961	0.9731	0.8464	1.0000
L_1 similarity	10	0.7413	0.7070	0.7977	0.9443	0.7359	0.9811	0.8151	0.9930
	20	0.7500	0.6777	0.8126	0.9756	0.4801	0.9956	0.8866	0.9920
	30	0.7626	0.6973	0.7981	0.9691	0.4801	0.9956	0.9164	0.9858
	40	0.7966	0.7074	0.7997	0.9726	0.5666	0.9955	0.9075	0.9846
	50	0.7825	0.6894	0.8289	0.9697	0.6211	0.9961	0.9011	0.9893
	60	0.8099	0.7021	0.8060	0.9690	0.7060	0.9949	0.8913	0.9879
L_2 similarity	10	0.7367	0.7140	0.8553	0.9214	0.6377	0.9070	0.7635	0.9852
	20	0.7452	0.7398	0.8168	0.9628	0.5079	0.9614	0.7978	0.9909
	30	0.7542	0.7632	0.8571	0.9651	0.5752	0.9699	0.8120	0.9854
	40	0.7614	0.7452	0.8443	0.9596	0.6280	0.9706	0.8133	0.9856
	50	0.7620	0.7363	0.8263	0.9635	0.6718	0.9720	0.8091	0.9860
	60	0.7521	0.7177	0.8337	0.9566	0.7006	0.9725	0.8110	0.9833
F-measure between 208-random-clusters of the 1040 samples and the ground truth clusters: 0.2043±0.0014 (based on 10 trials)									

Table 1 compares all the clustering results. With the spectral clustering, for all similarity scores, by appropriately enlarging the threshold to remove small entries in the similarity matrix, the noise is reduced. Consequently, the *F*-measure is improved significantly. For example, for correlation coefficient, when no thresholding is used, the respective *F*-measure (i.e. *F*(0)) is around 0.90~0.93; when thresholding using (μ+2σ), *F*-measure becomes larger than 0.99, indicating the clustering results have minor error. For both L_1 and L_2 similarities, the *F*-measure is also improved in the same way.

For MST-clustering, we constrained the in-cluster coherence using a factor β (the smallest similarity is no less than the largest similarity divided by β), so that within any obtained cluster there is no obvious outlier (i.e. very dissimilar document image). *F*-measures of the resultant clusters are shown in Table 1. It is clear that β has to be properly chosen. When β is too large (e.g. the case of $F(1.5)$), the obtained clusters could have poor coherence (e.g. low *F*-measures in L_1 and L_2 cases); when β is too small (e.g. the case of $F(1.05)$), an intrinsic cluster could be over split, leading to low *F*-measures (e.g. all the three similarities). Despite the fact that a carefully selected β for a particular MST might yield good results (e.g. β=1.5 for correlation coefficient similarity, and β=1.2 in the L_1 case), generally, most β values would not lead to promising clustering results.

In Table 1, the best results were obtained by combining MST clustering and spectral clustering. Clearly, for all similarity scores, the improvement is significant: the *F*-measure is consistently larger than 0.98. For correlation coefficient, *F*-measure is always 1, implying that there is no clustering error.

Table 1 also shows that the clustering results are not sensitive to the number of blocks used in generating CBP vectors. The 20 largest blocks appear to be sufficient to discriminate different form images. More blocks will not degrade the clustering. This observation is consistent to the classification results reported in [8] with the same data set.

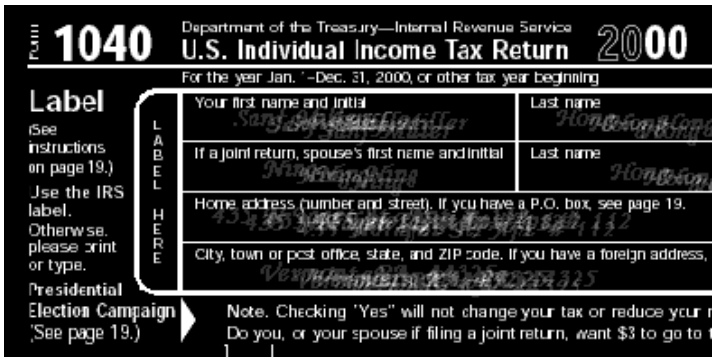
As a baseline comparison, we also calculated the *F*-measure between the 208 random clusters of the 1040 samples (5 per cluster) and the ground truth. The value is slightly larger than 0.2, which is in good accordance to our expectation.

We also generated smaller numbers of clusters. They have larger sizes, and might help to build hierarchical groups of form images. These results are omitted because they are not directly related to content extraction.

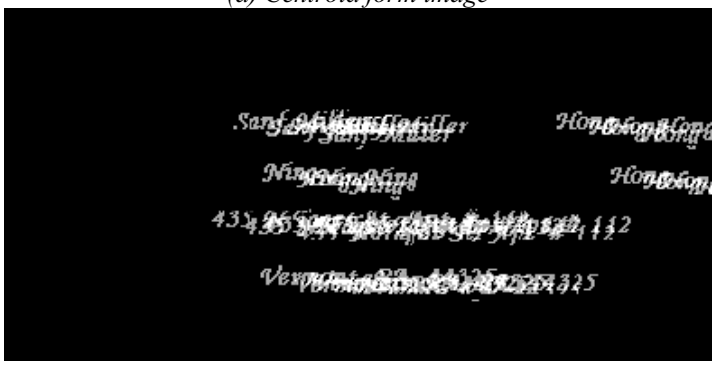
Table 1 also indicates choice of similarity scores: the correlation coefficient score gives the best overall *F*-measures, indicating the clusters found are most similar to the ground truth. The L_1 similarity is better than L_2 similarity, but not as good as correlation coefficient.

5.2. Template Construction and Content Extraction

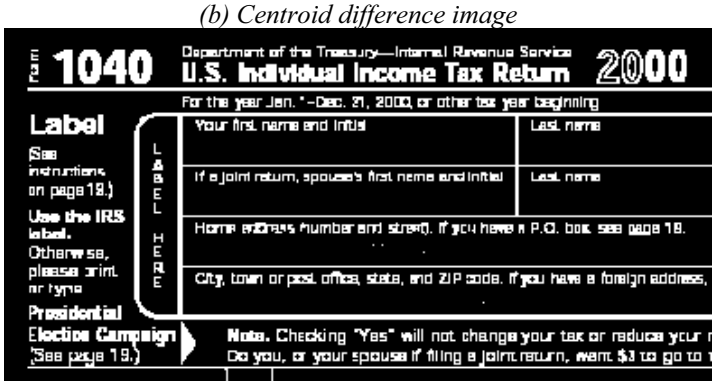
Based on the accurate clusters, we constructed the adaptive template images using the method shown in section 4. An example is included in Fig. 1. The CFI is shown in (a), where both the filled-in and the original information appear. The CDI is shown in (b), where only the input information appears. By combining (a) and (b) results, we constructed the non-filled form image in (c), which is the adaptive template image. Despite this clean example to show the basic idea, in a typical case the template is a bit fuzzy, which needs some post-processing (e.g. region based thinning, etc) to improve the quality. Finally, the filled-in content of each form was extracted by first aligning the filled form image with the template image, and then subtracting the template from the filled image. The result is shown in Fig.1 (d).



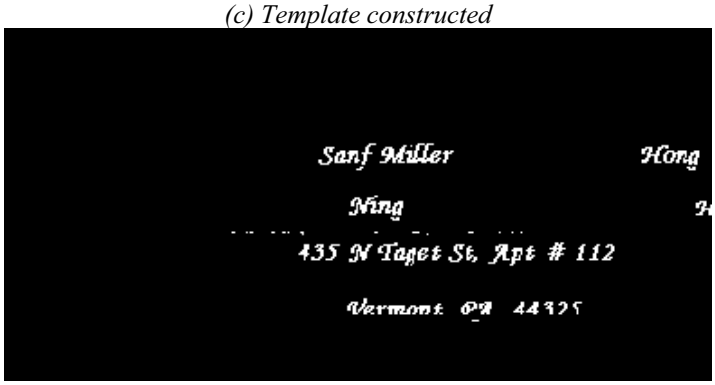
(a) Centroid form image



(b) Centroid difference image



(c) Template constructed



(d) The content extracted from a form image

Fig. 1 An example of filled-in content extraction.

6. DISCUSSIONS AND CONCLUSION

Clustering form images and building hierarchical groups is an intuitive idea in document image analysis. Nonetheless, this goal is difficult to achieve using traditional structural and semantic analysis methods. By analyzing the global layout information, highly discriminative features such as interval codes [5], pass-codes [6], and CBP vectors [8], can be obtained. Besides their usage in template matching (for comparison see [8]), they all have potentials in form image clustering.

We improved the spectral clustering by combining MST clustering. Clustering the MST using correlation coefficients leads to the best results, which are comparable to the earlier classification results in [8]. For other similarity scores, the new clustering scheme also improves the performance dramatically.

With the highly discriminative CBP features and the high-performance clustering scheme, the templates are no longer needed to be determined in advance. Alternatively, we adaptively constructed template images using the difference-form-images. This approach has been shown effective in extracting contents of a set of US tax form images.

ACKNOWLEDGEMENTS

We thank Drs. Sherry Li, Laura Grigori, and John Wu in Lawrence Berkeley National Lab for discussion on the related linear algebra problems.

APPENDIX: Precision, Recall, *E*-Measure, and *F*-Measure

Precision and recall are widely used terms in information retrieval [11]. Precision P defines proportion of retrieved materials that are relevant. Recall R defines the proportion of relevant materials that are retrieved. They have the following forms:

$$P = \frac{|\,\text{retrieved} \cap \text{relevant}\,|}{|\,\text{retrieved}\,|}, \tag{A1}$$

$$R = \frac{|\,\text{retrieved} \cap \text{relevant}\,|}{|\,\text{relevant}\,|}. \tag{A2}$$

Precision and recall can be combined as one score, weighted by a factor u [11]:

$$E = 1 - \frac{1}{\frac{u}{P} + \frac{1-u}{R}} = 1 - \frac{PR}{(1-u)P + uR}. \tag{A3}$$

The coefficient u has range [0,1], and can be equivalently written as $u = 1/(v^2+1)$, where v is unbounded. Hence, the score in Eq.(A3), called *E*-measure, has the following form,

$$E = 1 - \frac{(v^2+1)PR}{v^2 P + R}. \tag{A4}$$

Accordingly, the *F*-measure is defined as Eq.(A5), which reduces to the Eq.(4) when precision and recall are equally weighted, i.e. $v = 1$ or $u = 0.5$. Larger *F*-measure values are better.

$$F = 1 - E = \frac{(v^2 + 1)PR}{v^2 P + R}. \quad \text{(A5)}$$

REFERENCES

[1] Cesarini, F., Gori, M., Marinai, S., and Soda, G., "INFORMys: a flexible invoice-like form-reader system," IEEE Trans. PAMI, 20(7), pp.710-745, 1998.

[2] Corman, T.H., Leiserson, C.E., and Rivest, R.L., Introduction to Algorithms, MIT, 2001.

[3] Ding, C., He, X., Zha, H., Gu, M., and Simon, H., "A min-max cut algorithm for graph partitioning and data clustering," Proc. of 1st IEEE Int'l Conf. Data Mining. San Jose, CA, pp.107-114, 2001

[4] Fan, K., and Chang, M., "Form document identification using line structure based features," ICPR1998, vol.2, pp.1098-1100, 1998.

[5] Hu, J., Kashi, R., and Wilfong, G., "Document image layout comparison and classification," ICDAR1999, pp.285-288, Bangalore, India, 1999.

[6] Hull, J.J., "Document image similarity and equivalence detection," IJDAR, 1(1), pp.37-42, 1998.

[7] Peng, H., Chi, Z., Siu, W., and Feng, D., "PageX: an integrated document processing software for digital libraries," Proc of 2000 Int Workshop on Multimedia Data Storage, Retrieval, Integration, and Applications, Hong Kong, pp.203-207, 2000.

[8] Peng, H., Long, F., and Chi, Z., "Document image recognition based on template matching of component block projections," IEEE Trans. PAMI, 25(9), pp.1188-1192, 2003.

[9] Peng, H., Long, F., Chi, Z., and Siu, W., "Document template matching based on component block list," Pattern Recognition Letters, 22(9), pp.1033-1042, 2001.

[10] Pothen, A., Simon, H.D., and Liou, K.P., "Partitioning sparse matrices with eigenvectors of graph," SIAM Journal of Matrix Anal. Appl., 11, pp.430-452, 1990.

[11] van Rijsbergen, K. Information Retrieval, (2nd Ed.) (www.dcs.gla.ac.uk/Keith/Preface.html) Butterworths, London, 1979.

[12] Safari, R., Narasimhamurthi, N., Shridhar, M., and Ahmadi, M., "Document registration using projective geometry," IEEE Trans on Image Processing, vol.6, no.9, pp.1337-1341, 1997.

[13] Shi, J., and Malik, J., "Normalized cuts and image segmentation," IEEE Trans. PAMI, 2000.

[14] Shimotsuji, S., and Asano, M., "Form identification based on cell structure," 13th ICPR, vol.3, pp.793-797, 1996.

[15] Tseng, L., and Chen, R., "The recognition of form documents based on three types of line segments," ICDAR1997, 1, pp.71-75, 1997.

[16] Watanabe, T., Luo, Q., and Sugie, N., "Layout recognition of multi-kinds of table form documents," IEEE Trans. PAMI, 17(4), pp.432-445, 1995.